# Оглавление

[Оглавление 1](#_Toc63544767)

[Рейтинг 2](#_Toc63544768)

[ЛБ 1. Циклы 3](#_Toc63544769)

[ЛБ 2. Массивы 4](#_Toc63544770)

[ЛБ 3. Строки 5](#_Toc63544771)

[ЛБ 4. Условные операторы 5](#_Toc63544772)

[ЛБ 5. Модуль Math 6](#_Toc63544773)

[ЛБ 6. Алгоритмы 7](#_Toc63544774)

[ЛБ 7. Подпрограммы 8](#_Toc63544775)

[ЛБ 8. Форматирование 9](#_Toc63544776)

[ЛБ 9. Регулярные выражения 10](#_Toc63544777)

[ЛБ 10. Рекурсия 11](#_Toc63544778)

[ЛБ 11. Даты 12](#_Toc63544779)

[ЛБ 12. Классы, объекты 13](#_Toc63544780)

[ЛБ 13. Работа с DOM и HTTP 14](#_Toc63544781)

[ЛБ 14. Сортировка 15](#_Toc63544782)

[ЛБ 15. Дополнительная работа 16](#_Toc63544783)

**ВАЖНО!**

* Проверка исходных данных на корректное значение при решении всех задач **обязательна**!
* При работе со строками, **необходимо** учитывать наличие в ней всех допустимых букв, цифр и знаков препинания (если это особо не обговорено в условии).
* Везде, где исходные значения не заданы явно, **предусмотреть возможность их изменения** пользователем в процессе тестирования.
* По умолчанию, вывод результата должен производиться **через консоль, либо через элементы на веб-странице**.
* При выполнении ЛБ №15 можно использовать как «клиент-серверные» технологии, так и осуществлять реализацию полностью в браузере, используя предназначенные для этого средства (например, https://learn.javascript.ru/data-storage)
* Названия файлов отчётов, загружаемых в электронный курс, должны соответствовать шаблону: GGGG\_FIO\_NN, где GGGG – номер группы, FIO – фамилия и инициалы студента, NN – номер работы. Можно использовать как русский, так и английский языки. **Файлы с именами, отличающимися от вышеприведённого шаблона, приниматься НЕ будут, и баллы за них также начислены не будут!**

## Рейтинг

Общее число ЛБ – 15.

ЛБ №№ 1-14 оцениваются максимум в 5 баллов. ЛБ № 15 – максимум в 10 баллов.

Варианты на ЛБ №1 и № 15 выдаются согласно порядковому номеру студента (ПНС) в электронной ведомости. В случае, если ПНС больше 15, его вариант рассчитывается по формуле «ПНС-15». Варианты на ЛБ №№ 2-14 выдаются преподавателем по его усмотрению.  
Общая итоговая сумма баллов, полученная за ЛБ течение семестра, зависит от того, как сдавались работы (баллы могут быть снижены в процессе приёма при наличии недочётов) и соблюдения сроков сдачи (см. таблицы, в них приводится коэффициент успеваемости, на который будут умножаться баллы, полученные за каждую ЛБ).

**ОБРАТИТЕ ВНИМАНИЕ! Сдача любой ЛБ на «полные баллы» возможна только при условии, что для сдачи ЛБ использовалось не более одной попытки (за каждую последующую попытку баллы могут быть снижены)!**

Коэффициенты успеваемости для ЛБ №№ 1-14 (в зависимости от срока сдачи):

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ЛБ\неделя | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
| **1** | 1 | 0,9 | 0,8 | 0,7 | 0,6 | 0,5 | 0,4 | 0,3 | 0,2 | 0,1 | 0,1 | 0,1 | 0,1 | 0,1 | 0,1 | 0,1 |
| **2** | 1 | 1 | 0,9 | 0,8 | 0,7 | 0,6 | 0,5 | 0,4 | 0,3 | 0,2 | 0,1 | 0,1 | 0,1 | 0,1 | 0,1 | 0,1 |
| **3** | 1 | 1 | 1 | 0,9 | 0,8 | 0,7 | 0,6 | 0,5 | 0,4 | 0,3 | 0,2 | 0,1 | 0,1 | 0,1 | 0,1 | 0,1 |
| **4** | 1 | 1 | 1 | 1 | 0,9 | 0,8 | 0,7 | 0,6 | 0,5 | 0,4 | 0,3 | 0,2 | 0,1 | 0,1 | 0,1 | 0,1 |
| **5** | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,8 | 0,7 | 0,6 | 0,5 | 0,4 | 0,3 | 0,2 | 0,1 | 0,1 | 0,1 |
| **6** | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,8 | 0,7 | 0,6 | 0,5 | 0,4 | 0,3 | 0,2 | 0,1 | 0,1 |
| **7** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,8 | 0,7 | 0,6 | 0,5 | 0,4 | 0,3 | 0,2 | 0,1 |
| **8** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,8 | 0,7 | 0,6 | 0,5 | 0,4 | 0,3 | 0,2 |
| **9** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,8 | 0,7 | 0,6 | 0,5 | 0,4 | 0,3 |
| **10** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,8 | 0,7 | 0,6 | 0,5 | 0,4 |
| **11** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,8 | 0,7 | 0,6 | 0,5 |
| **12** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,8 | 0,7 | 0,6 |
| **13** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,8 | 0,7 |
| **14** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0,9 | 0,8 |

Коэффициенты успеваемости для ЛБ № 15 (в зависимости от срока сдачи):

|  |  |
| --- | --- |
| До конца 1-го месяца обучения | 1 |
| До конца 2-го месяца обучения | 0,9 |
| До конца 3-го месяца обучения | 0,8 |
| До конца 4-го месяца обучения | 0,7 |
| 5-й месяц и позже | 0,6 |

## ЛБ 1. Циклы

1. Создайте функцию, которая удалит буквы «a», «b» и «c» из данной строки и вернет измененную версию. Если данная строка не содержит «a», «b» или «c», вернуть null.
2. Напишите функцию, которая возвращает позицию второго вхождения "zip" в строку или -1, если оно не встречается как минимум дважды. Ваш код должен быть достаточно общим, чтобы передавать все возможные случаи, когда "zip" может встречаться в строке.
3. Изограмма - это слово, в котором нет повторяющихся букв. Создайте функцию, которая принимает строку и возвращает true или false в зависимости от того, является ли она «изограммой».
4. Создайте функцию, которая принимает массив целых чисел, отдельно суммирует четные и нечетные числа, а затем возвращает разницу между суммой четных и нечетных чисел.
5. Напишите функцию, которая принимает массив и возвращает новый массив с неуникальными отрицательными числами.
6. По заданному массиву чисел, представляющих высоту горы в определенных интервалах, определите, является ли эта гора масштабируемой. Гору можно считать масштабируемой, если каждое число находится в пределах 5 единиц от следующего числа в любом направлении.
7. Создайте функцию, которая принимает на вход число n и возвращает все числа до n включительно, соединенные в строку. Каждую цифру отделяйте друг от друга знаком «-».
8. Создайте функцию, которая принимает массив чисел arr и число n. Верните true, если сумма любых двух элементов равна заданному числу. В противном случае верните false.
9. Создайте функцию, которая возвращает общее количество шагов, необходимых для преобразования каждого элемента в максимальный элемент в массиве. Каждый шаг состоит из увеличения цифры на единицу.
10. Определите наименьшее количество шагов, необходимых для преобразования строки полностью в верхний регистр или полностью в нижний регистр, в зависимости от того, что занимает наименьшее количество шагов. Шаг состоит из изменения одного символа с нижнего на верхний регистр или наоборот.
11. Напишите функцию, которая возвращает самую длинную последовательность нулей в двоичной строке.
12. Представьте себе устройство для обмена сообщениями с одной кнопкой. Для буквы A вы нажимаете кнопку один раз, для буквы E вы нажимаете ее пять раз, для буквы G она нажимается семь раз, и т.д. Напишите функцию, которая принимает строку (сообщение) и возвращает общее количество раз, сколько кнопка нажата.
13. Создайте функцию, которая принимает символы строки как ASCII и возвращает шестнадцатеричное значение каждого символа как строку.
14. Создайте функцию, которая принимает целое число и выводит квадрат размером n x n, состоящий исключительно из целого числа n.
15. Предположим, что для каждого слоя ткани, который вы носите, когда на улице холодно (пальто, кардиганы, и т. д.), температура увеличивается на одну десятую от общей. Учитывая количество слоев и заданную температуру, определите температуру внутри всех этих слоев. Округлите ответ до ближайшей десятой градуса.

## ЛБ 2. Массивы

1. Напишите функцию, которая принимает координаты двух точек на двухмерной плоскости и возвращает длину отрезка, соединяющего эти две точки.
2. Предположим, изображение можно представить в виде двумерного массива нулей и единиц. Напишите функцию для создания «негатива» изображения. Замените 0 на 1 и наоборот.
3. Создайте функцию, которая возвращает количество истинных значений в массиве.
4. Напишите функцию, которая преобразует объект в массив, где каждый элемент представляет пару ключ-значение в форме массива.
5. Создайте функцию, которая принимает массив чисел и возвращает второе по величине число.
6. Создайте функцию, которая принимает массив (результат игрового автомата) и возвращает true, если все элементы в массиве идентичны, и false в противном случае. В массиве будет 4 элемента.
7. Учитывая массив ящиков, создайте функцию, которая возвращает общий объем всех этих ящиков, объединенных вместе. Ящик представлен ​​массивом из трех элементов: длины, ширины и высоты.
8. Массив является особенным, если каждый четный индекс содержит четное число, а каждый нечетный индекс содержит нечетное число. Создайте функцию, которая возвращает true, если массив особенный, и false в противном случае.
9. Создайте функцию, которая принимает массив элементов и проверяет, соответствует ли последний элемент остальной части объединенного вместе массива.
10. Создайте функцию, которая принимает массив слов и возвращает массив, в котором только первая буква каждого слова является заглавной.
11. Создайте функцию, которая принимает массив чисел и возвращает сумму двух наименьших положительных чисел.
12. Массивы можно смешивать с разными типами. Ваша задача в этой задаче - просуммировать все числовые элементы в данном массиве. Создайте функцию, которая принимает массив и возвращает сумму всех чисел в массиве.
13. Создайте функцию, которая принимает число и возвращает массив с цифрами числа в обратном порядке.
14. Создайте функцию, которая возвращает только строки с уникальными символами из массива строк.
15. Подсчитайте общее количество массивов внутри данного массива.

## ЛБ 3. Строки

1. Создайте функцию, которая принимает предложение и возвращает его аббревиатуру. Получите все слова длиной не менее n символов и верните первую букву каждого, с заглавной буквы и в целом, возвращенную как одну строку.
2. Создайте функцию, которая принимает два числа и математический оператор и возвращает результат.
3. Создайте функцию, которая возвращает true, если входная строка содержит только прописные или строчные буквы.
4. Дана строка чисел, разделенная запятой или пробелом, верните результат произведения чисел.
5. Создайте функцию, которая принимает строку и заменяет гласные другим символом.
6. Напишите функцию, которая меняет местами все слова в предложении, начинающиеся с определенной буквы.
7. Напишите функцию, которая принимает строку и прореживает ее (т.е. удаляет последнюю 1/10 символов). Если строка состоит из 21 символа, 1/10 символов будет равно 2,1 символа, поэтому функция удаляет 3 символа (округляем «вверх»)
8. На входе имеется массив: [[[[[[[[[[[]]]]]]]]]]]] (другими словами, массив, содержащий массив, содержащий массив, содержащий ... массив, не содержащий ничего). Ваша цель - измерить глубину этого массива, где [] имеет глубину 1, [[]] имеет глубину 2, [[[]]] имеет глубину 3, и т. д.
9. Создайте функцию, которая принимает слово и возвращает true, если слово имеет две идущие подряд одинаковые буквы.
10. Создайте функцию, которая из массива строк возвращает только строки с уникальными символами.
11. У вас есть массив кодов товаров в следующем формате: «[буквы] [цифры]». Создайте функцию, которая разбивает эти строки на их буквенные и числовые части.
12. Создайте функцию, которая принимает входные данные (например, «5 + 4») и возвращает true, если это математическое выражение, или false, если нет.
13. Создайте функцию, которая удалит любые повторяющиеся символы в слове, переданном в функцию (не только последовательные символы, но и символы, повторяющиеся в любом месте ввода).
14. Создайте функцию, которая добавляет пробелы перед каждой заглавной буквой в слове. После этого уберите заглавные буквы из всей строки.
15. Для строки, содержащей набор символов и цифр, вернуть сумму всех чисел в строке. Обратите внимание, что несколько цифр рядом друг с другом считаются как целое число, а не как отдельные цифры.

## ЛБ 4. Условные операторы

1. Создайте функцию, которая принимает три целочисленных аргумента (a, b, c) и возвращает количество целых чисел, имеющих равное значение.
2. Создайте функцию, которая принимает три значения: часы, минуты, секунды. Верните значение, которое является наибольшей продолжительностью.
3. Создайте функцию, которая принимает две строки (p1 и p2 - представляют игроков 1 и 2) в качестве аргументов и возвращает строку с указанием победителя в игре «камень, ножницы, бумага». Каждый аргумент будет содержать одну строку: «Камень», «Бумага» или «Ножницы».
4. Создайте функцию, которая принимает массив положительных и отрицательных чисел. Вернуть массив, в котором первый элемент - это количество положительных чисел, а второй - сумма отрицательных чисел.
5. Создайте функцию, которая принимает число в качестве входных данных и возвращает true, если сумма ее цифр имеет ту же четность, что и все число. В противном случае верните false.
6. Создайте функцию, которая изменяет все элементы в массиве следующим образом: добавляет 1 ко всем четным целым числам, 0 - к нечетным целым числам. Также добавляет "!" ко всем строкам и делает первую букву слова заглавной. Также изменяет все логические значения на противоположные.
7. Напишите функцию, которая берет строку, разбивает ее, и возвращает сначала гласные, а затем согласные. Любой символ, который не является гласным (например, специальные символы или пробелы), определяется как согласный.
8. При создании переменных, имя переменной всегда должно начинаться с буквы и не может содержать пробелов, хотя в нем также могут содержаться числа и символы подчеркивания. Создайте функцию, которая возвращает true, если данное имя переменной допустимо, иначе возвращает false.
9. Создайте функцию, которая принимает две числовые строки и возвращает их сумму в виде строки.
10. Создайте функцию, которая возвращает тип треугольника с учетом длин сторон. Верните следующие значения, если они соответствуют критериям: нет равных сторон - «разносторонний», две стороны равны - «равнобедренный», все стороны равны - «равносторонний», дано меньше или больше трех сторон - «не треугольник».
11. Напишите функцию, которая принимает массив из двух чисел и определяет, равны ли суммы цифр в каждом числе друг другу.
12. Проверьте, состоит ли данная строка только из букв и пробелов, и все ли буквы в нижнем регистре.
13. Напишите функцию, которая принимает строку и вычисляет количество букв и цифр в ней. Вернуть результат как объект.
14. Создайте функцию, которая принимает строку в качестве аргумента и возвращает эквивалент кода Морзе.
15. Создайте функцию, которая принимает массив и определяет, строго ли он возрастает, строго убывает, или нет.

## ЛБ 5. Модуль Math

1. Создайте функцию, которая принимает три параметра, где x - начало диапазона (включительно), y - конец диапазона (включительно), n - проверяемый делитель. Возвращает упорядоченный массив с числами в диапазоне, которые делятся на третий параметр n. Вернуть пустой массив, если нет чисел, делящихся на n.
2. Создайте функцию, которая принимает строку чисел, разделенных пробелами, и возвращает наибольшее и наименьшее число (в виде строки).
3. Напишите функцию, которая возвращает наибольший общий делитель (НОД) двух целых чисел.
4. Учитывая массив чисел и значение n, напишите функцию, которая возвращает вероятность выбора числа, большего или равного n, из массива. Вероятность должна быть выражена в процентах, округленных до одного десятичного знака.
5. Две матрицы должны иметь равное количество строк и столбцов для вычитания. В этом случае вычитаемая из двух матриц A и B будет матрицей, которая имеет такое же количество строк и столбцов, что и A и B. Результат вычитания A и B, обозначенный A - B, вычисляется путем вычитания соответствующих элементов A и B. Создайте функцию, которая принимает 2 x 2D-массив (матрицы A, B) в качестве аргумента и возвращает 2D-массив (matrix C).
6. Деление на 0 - ошибка, и ее следует избегать любой ценой. Создайте функцию, которая при задании математического выражения в виде строки возвращает true, если в любой точке выражение включает деление на 0.
7. Создайте функцию, которая принимает массив целых чисел и возвращает true, если каждое число является простым. В противном случае верните false.
8. По двум линиям определите, параллельны ли они. Строки представлены массивом [a, b, c], который соответствует линии ax + b.
9. Создайте функцию, которая принимает два вектора как массивы и проверяет, ортогональны ли эти два вектора или нет. Возвращаемое значение - логическое. Два вектора a и b ортогональны, если их скалярное произведение равно нулю.
10. Создайте функцию, которая принимает число n в качестве аргумента и проверяет, может ли данное число быть выражено как сумма двух или более последовательных положительных чисел.
11. Повторная цифра - это положительное число, составленное из одной и той же цифры. Создайте функцию, которая принимает целое число и возвращает, является ли это число повторной цифрой или нет.
12. Создайте функцию, которая принимает три аргумента a, b, c и возвращает сумму чисел, равномерно разделенных на c из диапазона a и b включительно.
13. Повторяющийся квадратный корень из числа - это количество раз, которое необходимо применить к функции извлечения квадратного корня, чтобы число было строго меньше 2. Для заданного целого числа вернуть повторяющийся квадратный корень. Вернуть «недействительно», если оно отрицательное.
14. Учитывая два целых числа a и b, верните, сколько раз a можно уменьшить вдвое, оставаясь при этом больше b.
15. Создайте функцию, которая принимает число a и находит недостающий показатель x так, чтобы a при возведении в степень x было равно b.

## ЛБ 6. Алгоритмы

1. Создайте функцию, возводящую в квадрат каждую цифру числа.
2. Создайте функцию, которая принимает массив чисел от 1 до 10 (исключая одно число) и возвращает отсутствующее число.
3. Создайте функцию, которая возвращает среднее значение всех цифр числа.
4. Создайте функцию, которая принимает массив слов и возвращает массив, отсортированный от самого короткого к самому длинному.
5. Создайте функцию, которая принимает массив чисел и возвращает уникальное число.
6. Создайте функцию, которая возвращает true, если заданные круговые области пересекаются, в противном случае возвращает false. Круги представлены в виде двух массивов, содержащих значения в следующем порядке: радиус круга, положение центра по оси x, положение центра по оси y.
7. Напишите функцию, которая возвращает наибольший общий делитель всех элементов массива. Если наибольший общий делитель равен 1, верните 1.
8. Для массива с нечетным числом элементов верните, будет ли шкала наклоняться «влево» или «вправо» в зависимости от суммы чисел. Шкала укажет направление наибольшей суммы. Если обе стороны равны, верните «сбалансировано».
9. Создайте функцию, которая принимает массив интервалов и возвращает, сколько интервалов перекрывается с заданной точкой. Интервал перекрывает конкретную точку, если точка существует внутри интервала или на границе интервала. Например, точка 3 перекрывается интервалом [2, 4] (находится внутри) и [2, 3] (находится на границе).
10. Напишите функцию, которая находит наибольшее число в массиве nums, которое также является четным. Если четного числа нет, верните -1.
11. Учитывая массив целых чисел, представляющих цвет каждого носка, определите, сколько пар носков совпадающих цветов существует. Например, есть 7 носков цветов [1, 2, 1, 2, 1, 3, 2]. Остается одна пара цвета 1 и одна пара цвета 2. Осталось три нечетных носка, по одному каждого цвета. Количество пар равно 2. Создайте функцию, которая возвращает целое число, представляющее количество подходящих пар доступных носков.
12. Для массива nums, где каждое целое число находится в диапазоне от 1 до 100, вернуть отсортированный массив, содержащий только повторяющиеся числа из данного массива nums.
13. Создайте функцию, которая принимает массив имен в формате «Имя Фамилия» (например, «Иван Петров») и возвращает массив этих имен, отсортированных по длине их фамилий. Если длина нескольких фамилий одинакова, выполните сортировку в алфавитном порядке по фамилии.
14. Создайте функцию, которая принимает в качестве аргумента массив, содержащий вложенные массивы. Каждый подмассив состоит из 2 элементов. Первый элемент - числитель, а второй элемент - знаменатель. Возвратите сумму дробей, округленную до ближайшего целого числа.
15. Создайте функцию, которая возвращает true, если есть хотя бы одно простое число в заданном диапазоне от n1 до n2 (включительно), иначе false.

## ЛБ 7. Подпрограммы

1. Создайте функцию, которая берет год и возвращает правильный век.
2. Создайте функцию, которая возвращает сумму всех четных элементов в 2D-матрице.
3. Создайте функцию, которая возвращает количество чисел палиндрома в указанном диапазоне (включительно). Например, между 8 и 34 имеется 5 палиндромов: 8, 9, 11, 22 и 33. Между 1550 и 1552 годами существует ровно один палиндром: 1551.
4. В шахматах ферзи могут перемещаться на любое количество клеток по горизонтали, вертикали или диагонали. Учитывая расположение вашего ферзя и ферзя оппонента, определите, можете ли вы захватить ферзя оппонента. Ваше местоположение и местоположение ваших противников - это первый и второй элементы массива соответственно.
5. Напишите программу для фильтрации неуникальных значений в массиве на основе предоставленной функции сравнения.
6. Напишите программу, чтобы проверить, является ли строка анаграммой другой строки (без учета регистра, игнорирует пробелы, знаки препинания и специальные символы).
7. Напишите программу для добавления порядкового суффикса к предложению.
8. Напишите программу для генерации всех перестановок строки (может содержать дубликаты).
9. Создайте функцию, которая выполняет преобразование четно-нечетное в массив n раз. Каждое четное преобразование добавляет два (+2) к каждому нечетному целому числу, каждое нечетное вычитает два (-2) из каждого четного числа.
10. Создайте функцию, которая инвертирует значения RGB данного кортежа (примеры: [0, 0, 0]) ➞ [255, 255, 255], [165, 170, 221]) ➞ [90, 85, 34].
11. Вы работаете на производителя, и вас попросили подсчитать общую прибыль, полученную от продажи продукта. Вам предоставляется объект, содержащий себестоимость единицы, цену продажи за единицу и начальные запасы. Вернуть общую полученную прибыль, округленную до ближайшего целого.
12. Учитывая две строки, создайте функцию, которая возвращает общее количество уникальных символов из объединенной строки.
13. Создайте функцию, которая удалит любые повторяющиеся символы в слове, переданном в функцию (не только последовательные символы, но и символы, повторяющиеся в любом месте ввода).
14. Создайте функцию, которая возвращает количество символов, общих для двух слов.
15. Для объекта, содержащего имена и возрасты группы людей, верните имя самого старого человека.

## ЛБ 8. Форматирование

1. Создайте функцию, которая принимает число как строку и возвращает число без конечных и ведущих нулей.
2. Создайте функцию, которая перемещает все заглавные буквы в начало слова.
3. Подсчитайте количество единиц в двоичном представлении целого числа. Например, поскольку 12 - это 1100 в двоичном формате, возвращаемое значение должно быть равно 2.
4. Напишите функцию, которая принимает номер кредитной карты и отображает только последние четыре символа. Оставшуюся часть номера карты необходимо заменить на \*\*\*\*\*\*\*\*\*\*\*\*.
5. Создайте функцию, которая принимает строку и вставляет тире слева и справа от каждой гласной.
6. Получив строку, создайте функцию для изменения регистра. Все буквы в нижнем регистре должны быть в верхнем регистре, и наоборот.
7. Создайте функцию, которая принимает строку и возвращает средний символ (символы). Если длина слова нечетная, верните средний символ. Если длина слова четная, вернуть два средних символа.
8. Создайте функцию, которая принимает массив из 10 чисел (от 0 до 9) и возвращает строку этих чисел, отформатированных как номер телефона (например, (555) 555-5555).
9. Создайте функцию, которая принимает число в качестве аргумента и возвращает строку, отформатированную для разделения тысяч.
10. Напишите функцию, которая принимает в качестве аргумента строку из одного или нескольких слов и возвращает ту же строку, но с перевернутыми словами из пяти или более букв. Переданные строки будут состоять только из букв и пробелов. Пробелы будут включены только в том случае, если присутствует более одного слова.
11. Напишите функцию, которая принимает массив и возвращает новый массив с уникальными положительными числами.
12. К предложению добавлены дополнительные пробелы. Верните правильное предложение, удалив их. Все слова должны быть разделены одним пробелом, и в начале и в конце предложения не должно быть пробелов.
13. Создайте функцию, которая принимает последовательность строк или чисел, удаляет окружающие дубликаты и возвращает массив элементов без каких-либо элементов с одинаковым значением рядом друг с другом, также сохраняет исходный порядок элементов.
14. Создайте функцию, которая принимает одну строку в качестве аргумента и возвращает упорядоченный массив, содержащий индексы всех заглавных букв в строке.
15. Создайте функцию, которая делает последнюю букву каждого слова заглавной.

## ЛБ 9. Регулярные выражения

1. Банкоматы допускают использование 4- или 6-значных PIN-кодов, а PIN-коды не могут содержать ничего, кроме ровно 4 или 6 цифр. Ваша задача - создать функцию, которая принимает строку и возвращает true, если ПИН-код действителен, и false, если нет.
2. Создайте функцию, которая берет слово и меняет все гласные на число.
3. Ваша задача - создать регулярное выражение для поиска ссылки на Twitter. Это регулярное выражение просматривает твит, чтобы найти «@handle» и «#handle». Возвращайте только слова без дескрипторов @ и #.
4. Создайте функцию, которая принимает массив имен супергероев и возвращает массив только имен супергероев, начинающихся на «человек». Верните имена в алфавитном порядке.
5. Создайте функцию, которая принимает массив из двух строк и проверяет, присутствуют ли буквы из второй строки в первой строке.
6. Создайте регулярное выражение, чтобы проверить, является ли данная строка допустимым числовым символом с плавающей запятой.
7. Создайте функцию, которая возвращает true, если данная строка имеет любое из следующих значений: только буквы и никакие числа, только цифры и никаких букв. Если строка содержит и числа, и буквы, или символы, не попадающие ни в одну категорию, верните false.
8. Проверьте, состоит ли данная строка только из букв и пробелов, и все ли буквы в нижнем регистре.
9. Создайте функцию, которая принимает строку, содержащую целые числа, а также другие символы, и возвращает только сумму отрицательных целых чисел.
10. Создать функцию, которая преобразует предложения, заканчивающиеся множеством вопросительных знаков «?» или восклицательных знаков «!» в предложение, оканчивающееся только на один знак, без изменения знаков препинания в середине предложения.
11. Напишите функцию, которая принимает строку и вычисляет количество букв и цифр в ней. Вернуть результат как объект.
12. Напишите регулярное выражение, которое ищет все десятичные числа, включая целые, с плавающей запятой и отрицательными значениями.
13. Создайте функцию, которая принимает строку слов (или только одно слово) и преобразует каждое слово из camelCase в snake\_case.
14. Создайте функцию, которая подсчитывает количество блоков из двух или более смежных единиц в массиве.
15. Напишите функцию, которая извлекает все слова, начинающиеся с гласной.

## ЛБ 10. Рекурсия

1. Напишите функцию, которая вычисляет n-е число Фибоначчи.
2. Напишите функцию, которая переворачивает строку. Сделайте вашу функцию рекурсивной.
3. Напишите функцию, которая рекурсивно вычисляет НОД (наибольший общий делитель) двух чисел.
4. Напишите функцию, которая находит сумму первых n натуральных чисел. Сделайте вашу функцию рекурсивной.
5. Создайте функцию, которая вычисляет количество различных квадратов в квадратной сетке n \* n.
6. Создайте функцию, которая находит наибольшее целое число в массиве с помощью рекурсии.
7. Создайте рекурсивную функцию, которая принимает два параметра и повторяет строку n раз. Первый параметр txt - это строка, которую нужно повторить, а второй параметр - это количество раз, которое строка должна быть повторена. String.prototype.repeat() не допускается.
8. Создайте функцию, которая принимает число num и возвращает его двойной факториал.
9. Создайте функцию, которая принимает массив и возвращает сумму всех элементов в массиве.
10. Напишите функцию, которая рекурсивно определяет, является ли строка палиндромом.
11. Напишите функцию, рекурсивно возвращающую количество гласных в строке.
12. Создайте функцию, которая принимает положительное целое число (одно из base2, base8 или base16), преобразует целое число в заданное основание и возвращает строку с использованием рекурсии.
13. Учитывая известное количество уникальных предметов, сколькими способами мы можем расположить их в ряд? Создайте функцию, которая принимает целое число n и возвращает количество цифр числа возможных перестановок для n уникальных элементов. Например, 5 уникальных предметов можно расположить 120 уникальными способами. 120 состоит из 3 цифр, поэтому возвращается целое число 3.
14. Напишите функцию, которая находит наибольшее число в массиве nums, которое также является четным. Если четного числа нет, верните -1.
15. Создайте функцию, которая принимает многомерный массив, преобразует его в одномерный массив, и возвращает его с помощью рекурсии.

## ЛБ 11. Даты

1. Создайте функцию, которая принимает на вход год и месяцы, а затем вернет, какой год будет по прошествии n месяцев.
2. Создайте функцию, которая преобразует дату в формате ММ / ДД / ГГГГ в ГГГГДДММ.
3. Напишите функцию, которая, учитывая дату (в формате ММ / ДД / ГГГГ), возвращает день недели в виде строки. Название каждого дня должно быть одной из следующих строк: «Понедельник», «Вторник», «Среда», «Четверг», «Пятница», «Суббота», «Воскресенье».
4. Создайте функцию, которая принимает месяц и год (как целые числа) и возвращает количество дней в этом месяце.
5. Если месяц и год представлены в виде чисел, узнайте, содержит ли этот месяц пятницу, 13-е.
6. Учитывая параметры день, месяц и год, верните, является ли эта дата допустимой датой.
7. Для данной даты верните, сколько дней до следующего года. Дата будет в формате ММ / ДД / ГГГГ.
8. 2 февраля 2020 года - это палиндромная дата в формате ДД / ММ / ГГГГ и ММ / ДД / ГГГГ (02.02.2020). Для входной даты в формате ДД / ММ / ГГГГ вернуть true, если дата палиндромная в обоих форматах даты, в противном случае вернуть false.
9. Создайте функцию, которая преобразует 12-часовое время в 24-часовое или наоборот. Вернуть результат в виде строки.
10. Создайте функцию, которая принимает две даты и возвращает количество дней между первой и второй датами.
11. Создайте функцию, которая возвращает количество пятниц, 13-го числа в заданном году.
12. Создайте функцию для расчета счета за электроэнергию. Учитывая дату начала и окончания выставления счета, начальные и конечные показания счетчика, цену за единицу в рублях и постоянную плату (ежедневная арендная плата за ваш счетчик) в качестве аргументов, рассчитайте свой счет. Счет за электроэнергию рассчитывается путем умножения разницы между показаниями счетчика на цену за единицу и добавления количества дней, умноженного на постоянную плату. Затем вам нужно добавить 5% налога.
13. Учитывая диапазон лет в виде строки, верните количество високосных лет в пределах диапазона (включительно).
14. Для объекта Date () верните дату за три дня до этого в виде строки: «2016-01-19».
15. Если бы сегодня был понедельник, через два дня это была бы среда. Создайте функцию, которая принимает в качестве входных данных массив дней и количество дней для увеличения, и возвращает массив дней по прошествии n дней.

## ЛБ 12. Классы, объекты

1. Создайте метод в классе Person, который возвращает сравнение возраста другого человека. Учитывая экземпляры p1, p2 и p3, которые будут инициализированы атрибутами name и age, верните предложение в следующем формате: {имя другого человека} {старше / моложе / того же возраста}, что и я.
2. Получив число, выясните, сколько единиц, троек и девяток вы можете уместить в это число. Вы должны создать класс. Для этого вы создадите переменные (class.ones, class.threes, class.nines).
3. Напишите программу для сортировки объектов по значениям одного из полей.
4. Создайте полное имя и адрес электронной почты для свойств экземпляра в классе Employee. Учитывая имя и фамилию человека, сформируйте полное имя, просто соединив имя и фамилию, разделенные пробелом. Сформируйте адрес электронной почты, соединив имя и фамилию вместе с расширением «.» между ними и в конце укажите «@company.com». Убедитесь, что адрес написан строчными буквами.
5. Напишите класс с именем Name и создайте следующие атрибуты по имени и фамилии (как fname и lname): атрибут с именем fullname, который возвращает имя и фамилию; атрибут инициалов, который возвращает первые буквы имени и фамилии. Поместите расширение «.» между двумя буквами. Не забудьте также разрешить доступ к атрибутам fname и lname по отдельности.
6. Создайте функцию, которая принимает массив экземпляров из класса IceCream и возвращает значение сладости самого сладкого мороженого. Сладость рассчитывается на основе вкуса (числовой коэффициент) и количества посыпок.
7. Создавайте функции для класса Calculator, которые могут делать следующее: складывать два числа, вычитать два числа, умножать два числа, делить два числа.
8. Напишите программу для сравнения двух объектов, чтобы определить, содержит ли первый объект эквивалентные значения свойств для второго.
9. Напишите программу, чтобы получить отсортированный массив объектов, упорядоченных по именам свойств.
10. Напишите программу, которая назначает значения по умолчанию для всех свойств объекта, которые не заданы.
11. Напишите программу для записи объекта в JSON-файл.
12. Напишите программу для определения заданного значения во вложенном объекте JSON на основе заданного ключа.
13. Напишите программу для создания объекта, состоящего из свойств, для задания значений которых используется функция. Функция вызывается с двумя аргументами: (значение, ключ).
14. Напишите программу для создания нового объекта из комбинаций 2-х или более объектов.
15. Напишите программу, чтобы фильтровать массив объектов на основе условий (по значению ключа), а также отфильтровывать неопределенные ключи.

## ЛБ 13. Работа с DOM и HTTP

1. Измените стиль текста абзаца на странице через код (обработчик нажатия на кнопку).
2. Напишите функцию, чтобы получить значения имени и фамилии, взятой из формы.
3. В HTML-файле имеется ссылка. Напишите функцию, чтобы получить значения атрибута href указанной ссылки, и её текст.
4. Напишите функцию для добавления строк в HTML-таблицу.
5. Напишите функцию, которая принимает строку, столбец (для идентификации конкретной ячейки), строку для обновления содержимого этой ячейки в HTML-таблице, и затем обновляет её.
6. Напишите программу для удаления элементов из выпадающего списка.
7. Напишите программу для подсчета и отображения элементов раскрывающегося списка.
8. Напишите программу для отображения случайного изображения (нажав на кнопку) из списка.
9. Напишите программу, чтобы выделить жирным слова абзаца, наведя указатель мыши на определенную ссылку.
10. Напишите программу, чтобы получить ширину и высоту окна (каждый раз, когда размер окна изменяется).
11. Напишите программу, которая будет возвращать true, если родительский HTML-элемент содержит дочерний HTML-элемент, в противном случае - false.
12. Напишите программу, чтобы получить объект, содержащий параметры текущего URL.
13. Напишите программу, которая будет возвращать true, если фокус находится на странице во вкладке браузера, иначе false.
14. Напишите функцию, чтобы проверить, является ли данное значение допустимым URL-адресом или нет.
15. Напишите программу, которая будет возвращать значение true, если видна нижняя часть страницы, в противном случае - значение false.

## ЛБ 14. Сортировка

1. Создайте функцию, которая принимает массив массивов с числами. Верните новый (одиночный) массив с наибольшим числом каждого из них.
2. Вам будет предоставлен набор напитков, каждый из которых является объектом с двумя свойствами: названием и ценой. Создайте функцию с массивом напитков в качестве аргумента и верните объекты напитков, отсортированные по цене в порядке возрастания.
3. Учитывая массив из 10 чисел, верните максимально возможную сумму, полученную путем суммирования всего 5 из 10 чисел.
4. Напишите функцию, которая принимает строку в качестве аргумента и возвращает крайнюю левую цифру в строке.
5. Напишите функцию, которая принимает в качестве аргумента строку из одного или нескольких слов и возвращает ту же строку, но с перевернутыми словами из пяти или более букв. Переданные строки будут состоять только из букв и пробелов. Пробелы будут включены только в том случае, если присутствует более одного слова.
6. Создайте функцию сортировки, которая сортирует числа не по порядку номеров, а по длине. Это означает сортировку чисел с наименьшим количеством цифр перед числами с наибольшим количеством цифр.
7. Для заданного числа верните разницу между максимальным и минимальным числами, которые могут быть образованы при перестановке цифр.
8. Для массива целых чисел вернуть наибольший разрыв между элементами отсортированной версии этого массива.
9. Создайте функцию, которая принимает строку и возвращает true, если сумма позиций каждой буквы в алфавите четная, и false, если сумма нечетная.
10. Учитывая два массива, объедините их в один массив и отсортируйте новый массив в том же порядке, что и первый массив.
11. Для входной строки поменяйте местами строку слово за словом, первое слово будет последним и так далее.
12. У числа 609 есть особое свойство: если вы перевернете число (представьте, что вы повернёте экран на 180 градусов), вы снова получите 609. Напишите функцию, которая принимает строку с цифрами 0, 6, 9 и возвращает true, если число совпадает с перевернутым числом, или false в противном случае.
13. Напишите функцию, которая сортирует массив символов в порядке возрастания (от «a» до «z»).
14. Создайте функцию, которая принимает строку слов и возвращает строку, отсортированную в алфавитном порядке по последнему символу каждого слова.
15. Создайте функцию, которая определяет, можно ли переупорядочить элементы в массиве, чтобы сформировать последовательный список чисел, где каждое число появляется ровно один раз.

## ЛБ 15. Дополнительная работа

1. Разработка функционала приложения для торгового предприятия
2. Разработка функционала приложения для кредитного отдела банка
3. Разработка функционала приложения для гостиницы
4. Разработка функционала приложения для авторемонтной мастерской
5. Разработка функционала приложения для автосалона
6. Разработка функционала приложения для агентства недвижимости
7. Разработка функционала приложения для склада
8. Разработка функционала приложения для учета расходов семьи
9. Разработка функционала приложения для организации делопроизводства
10. Разработка функционала приложения для рекламного агентства
11. Разработка функционала приложения для службы поддержки
12. Разработка функционала приложения для кадровой службы организации
13. Разработка функционала приложения для туристического агентства
14. Разработка функционала приложения для компьютерного магазина
15. Разработка функционала приложения для страховой компании
16. Разработка функционала приложения для кафедры вуза
17. Разработка функционала приложения для мебельного магазина
18. Разработка функционала приложения для книжного магазина
19. Разработка функционала приложения для учета договоров в организации
20. Разработка функционала приложения для кафе
21. Разработка функционала приложения по взаимодействию с клиентами организации услуг
22. Разработка функционала приложения для взаимодействия с заказчиками в товарной организации
23. Разработка функционала приложения для салона красоты
24. Разработка функционала приложения сервисного центра по ремонту компьютерной техники
25. Разработка функционала приложения для транспортной компании
26. Разработка функционала приложения для менеджера по продажам
27. Разработка функционала приложения для учета компьютерной техники и ПО в организации
28. Разработка функционала приложения для организации документооборота
29. Разработка функционала приложения для управляющей компании ЖКХ
30. Разработка функционала приложения для салона сотовой связи